DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof. Reference herein to any social initiative (including but not limited to Diversity, Equity, and Inclusion (DEI); Community Benefits Plans (CBP); Justice 40; etc.) is made by the Author independent of any current requirement by the United States Government and does not constitute or imply endorsement, recommendation, or support by the United States Government or any agency thereof.

ORNL/SPR-2025/4188

Unrolled Video Super-Resolution Network with Autoregressive Prior for the Case of Known Motion

Haley Duba-Sullivan
Emma J. Reid
Charles A. Bouman
Gregery T. Buzzard

September 2025

OAK RIDGE National Laboratory

ORNL IS MANAGED BY UT–BATTELLE LLC FOR THE US DEPARTMENT OF ENERGY

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

ORNL/SPR-2025/4188

Cyber Resilience and Intelligence Division

UNROLLED VIDEO SUPER-RESOLUTION NETWORK WITH AUTOREGRESSIVE PRIOR FOR THE CASE OF KNOWN MOTION

Haley Duba-Sullivan
Emma J. Reid
Charles A. Bouman
Gregery T. Buzzard

September 2025

Prepared by
OAK RIDGE NATIONAL LABORATORY
Oak Ridge, TN 37831
managed by
UT-BATTELLE LLC
for the
US DEPARTMENT OF ENERGY
under contract DE-AC05-00OR22725

CONTENTS

LIST OF FIGURES iv
LIST OF ABBREVIATIONS vi
ABSTRACT 1
1. INTRODUCTION 1
2. RELATED WORK 3
2.1 VSR METHODS 3
2.2 UNROLLED METHODS 4
3. METHOD 6
3.1 FORWARD MODEL 6
3.2 AUTOREGRESSIVE UNROLLED OPTIMIZATION 6
3.3 DEEP UNROLLED NETWORK 8
3.3.1 DATA-FITTING MODULE 8
3.3.2 AUTOREGRESSIVE PRIOR MODULE 8
3.3.3 HYPERPARAMETER ESTIMATION MODULE 9
3.4 END-TO-END TRAINING 10
4. EXPERIMENTAL RESULTS 11
4.1 IMPACT OF NUMBER OF PNP ITERATIONS 11
4.2 IMPACT OF NUMBER OF LOW-RESOLUTION FRAMES 11
4.3 IMPACT OF NUMBER OF PREVIOUS FRAMES 12
4.4 ANALYSIS OF DATA-FITTING AND DENOISER MODULES 14
4.5 ANALYSIS OF HYPERPARAMETER ESTIMATION MODULE 16
4.6 COMPARISON TO OTHER SUPER-RESOLUTION METHODS 16
5. FUTURE WORK 23
6. CONCLUSION 24

LIST OF FIGURES

Figure 1. Overall pipeline of UVSRNet for super-resolving frame i. The network alternates between a data-fitting module, which enforces consistency with the observed LR frames, and an autoregressive denoiser, which serves as a learned prior. A hyperparameter estimation module (from USRNet [1]) adaptively predicts the iteration-specific hyperparameters that balance these two components. 8

Figure 2. Architecture of the autoregressive prior module. At each iteration, the denoiser refines the intermediate solution z_k by processing the concatenation of z_k, a noise-level map, and registered neighboring frames. DRUNet [43] serves as the backbone, but we extend the input layer to handle multiple frames for exploiting temporal correlations in video super-resolution (VSR). . 9

Figure 3. Quantitative evaluation of the impact of the number of PnP iterations K on reconstruction quality. Results are averaged over 15 simulated test videos with 25 frames each. Using $K = 7$ achieves the best performance, though differences across $K = 3, 5, 7, 9$ are relatively small. Note the limited y-axis ranges (PSNR: 28–30 dB, SSIM: 0.75–0.85). 12

Figure 4. Qualitative comparison of reconstructions using different numbers of PnP iterations K. Results are shown for a 200×200 crop of a test video. Cropped regions show minimal visible differences across $K = 3, 5, 7, 9$, consistent with the small quantitative variations shown in Figure 3. 13

Figure 5. Quantitative evaluation of the impact of the number of LR frames w on reconstruction quality. Results are averaged over 15 simulated test videos with 25 frames each. Incorporating neighboring frames ($w > 0$) leads to a significant improvement over the single-frame case ($w = 0$), highlighting the benefit of exploiting subpixel motion. Among the tested settings, using 5 LR frames ($w = 2$) achieves the best overall performance, though the difference between $w = 1$ and $w = 2$ is modest. Note the limited y-axis ranges (PSNR: 25–30 dB, SSIM: 0.65–0.85). 14

Figure 6. Qualitative comparison of reconstructions using different numbers of LR frames w. Results are shown for a 200×200 crop of a test video. Using $w > 0$ produces visibly sharper textures and reduces residual error compared to the single-frame case ($w = 0$). The difference between $w = 1$ and $w = 2$ is minor, consistent with the quantitative results in Figure 5. 15

Figure 7. Quantitative evaluation of the impact of the number of previous frames p on reconstruction quality. Results are averaged over 15 simulated test videos with 25 frames each. Using $p = 1$ achieves the best performance, though differences across $p = 0, 1, 2$ are relatively small. Note the limited y-axis ranges (PSNR: 28–30 dB, SSIM: 0.75–0.85). 16

Figure 8. Qualitative comparison of reconstructions using different numbers of previous frames p. Results are shown for a 200×200 crop of a test video. Differences among $p = 0, 1, 2$ are visually small, consistent with the quantitative results. 17

Figure 9. (a) Outputs of the data-fitting module (top row) and autoregressive denoiser (bottom row) across iterations for 4× super-resolution, with green box indicating the cropped region. (b) Magnified view of the cropped area, showing that the data-fitting module sharpens structures while the denoiser suppresses artifacts; their alternation yields higher-fidelity reconstructions. . . 18
Figure 10. Output of the hyperparameter module under 4× super-resolution forward model with AWGN of standard deviation $\sigma = 0.01$. Both the data-fitting step size and the denoiser noise level decrease across iterations, reflecting HQS-like convergence from coarse to fine updates. The data-fitting step size remains consistently larger, indicating stronger weighting on data fidelity relative to regularization, thereby balancing reconstruction accuracy with artifact suppression. 19
Figure 11. Quantitative evaluation of the reconstruction quality from USRNet, RealBasicVSR, and our proposed UVSRNet. Results are averaged over 15 simulated test videos with 25 frames each. UVSRNet achieves the best performance by a large margin. Note the limited y-axis ranges (PSNR: 19.5–30 dB, SSIM: 0.45–0.85). 19
Figure 12. Qualitative comparison of UVSRNet with USRNet and RealBasicVSR. The first row provides context with the LR, bicubic, and ground-truth HR frames, while the second row shows the reconstructions from the three methods. The residual maps in the third row highlight differences with respect to the ground truth. UVSRNet produces sharper edges, better preserves fine details, and yields smaller residuals compared to the baselines. 21
Figure 13. Zoomed-in comparison on a 150 × 150 region from Figure 12. UVSRNet more faithfully reconstructs fine structures—such as the thin vertical bars of the purple fence—that are lost or blurred in USRNet and RealBasicVSR. . 22

LIST OF ABBREVIATIONS

ADMM	alternating direction method of multipliers
AWGN	additive white gaussian noise
CGM	conjugate gradient method
CNNs	convolutional neural networks
HQS	half-quadratic splitting
HR	high-resolution
i.i.d.	independent and identically distributed
ISTA	iterative shrinkage-thresholding algorithm
LR	low-resolution
PnP	plug-and-play
PSNR	peak signal-to-noise ratio
SISR	single image super-resolution
SSIM	structural similarity index measure
VSR	video super-resolution

ACKNOWLEDGEMENTS

This manuscript has been authored in part by UT-Battelle, LLC, under contract DE-AC05-00OR22725 with the US Department of Energy (DOE). The publisher acknowledges the US government license to provide public access under the DOE Public Access Plan (https://energy.gov/doe-public-access-plan).

ABSTRACT

Real-time detection and classification of distant objects is necessary for many national security applications. However, when objects are far from the sensor, they occupy only a small number of pixels in the captured video, limiting the amount of visual detail available for recognition. State-of-the-art classification methods typically rely on high-resolution (HR) video streams to capture characteristic object features, but obtaining such detail is challenging for distant objects that occupy only a few pixels. This motivates the development of video super-resolution (VSR) methods that enhance object classification by recovering fine details from low-pixel representations. Current VSR methods rely either on model-based optimization, which is interpretable but computationally expensive, or on learning-based approaches, which are efficient and high-performing but often lack flexibility and interpretability.

In this report, we propose an end-to-end trainable unrolled VSR network, UVSRNet, which super-resolves each frame in a video by exploiting sub-pixel motion between neighboring low-resolution (LR) frames as well as incorporating high-frequency detail from previously super-resolved frames. In particular, by unrolling a plug-and-play (PnP) half-quadratic splitting (HQS) algorithm, we leverage a model-based data-fitting module alongside a learning-based autoregressive prior module. This combination yields a method that maintains the flexibility and interpretability of model-based methods while achieving the performance advantages of learning-based methods.

1. INTRODUCTION

The ability to detect and classify distant objects in real time is essential for many national security applications. Surveillance, reconnaissance, and threat monitoring systems must make decisions under constrained conditions, where accurate recognition is critical for operational success. A central challenge arises when objects of interest are far from the sensing platform. At such distances, targets occupy only a handful of pixels in the captured video stream, severely limiting the visual detail available for recognition. The scarcity of discriminative features in LR video directly impacts classification accuracy and reliability.

Conventional object-recognition methods are typically designed with the assumption that inputs contain rich HR detail. For example, deep convolutional neural networks (CNNs) rely on the presence of characteristic texture and shape cues to distinguish between object categories. While these methods perform well when applied to high-quality imagery, they significantly degrade in scenarios where objects span only a few pixels. In real-world deployments, the achievable imaging resolution is fundamentally limited by sensor hardware, meaning that beyond a certain distance, it is impossible to capture HR detail regardless of bandwidth or environmental conditions. This creates a pressing need for techniques that can enhance LR video streams to recover sufficient detail for accurate recognition.

VSR has emerged as a promising solution to this problem. By leveraging temporal information across multiple frames, VSR methods reconstruct HR video from LR inputs, effectively recovering fine-grained details that are otherwise absent. Existing approaches to VSR can be broadly grouped into two categories. Model-based optimization methods explicitly encode image formation and motion models into iterative algorithms through modeling of the physical system. These methods offer interpretability and the ability to incorporate domain knowledge, but they are often computationally intensive and difficult to scale for real-time applications. In contrast, learning-based approaches use deep neural networks to directly map LR inputs to HR outputs. Such methods are computationally efficient and deliver strong empirical performance, but they tend to operate as black boxes, lacking the flexibility and interpretability required in many critical applications. Unrolled and PnP frameworks offer a promising middle ground by embedding data-consistency updates alongside learnable priors, combining theoretical interpretability with strong empirical performance. However, the only existing unrolled VSR framework operates primarily in the LR domain, and its algorithmic formulation lacks rigorous theoretical justification (discussed in Section 2). This reveals a critical gap: current VSR research lacks architectures that are both effective and efficient, while remaining grounded in an interpretable and flexible optimization framework that can operate effectively within the inherent resolution constraints of real-world sensors.

To address this research gap, we propose UVSRNet, an end-to-end trainable, unrolled VSR network inspired by the single-image, unrolled super-resolution framework USRNet [1]. Like USRNet, UVSRNet is built upon the unrolling of a PnP HQS algorithm, which naturally integrates both model-based and learning-based components. However, whereas USRNet focuses on single image super-resolution (SISR), UVSRNet extends the formulation to video by explicitly leveraging sub-pixel motion between neighboring LR frames and incorporating an autoregressive prior. Specifically, our architecture combines a data-fitting module that enforces consistency with LR observations with an autoregressive prior module that contributes high-frequency detail from past reconstructions. Through this design, UVSRNet maintains the interpretability and flexibility of model-based methods while achieving the efficiency and reconstruction quality of learning-based networks.

The key contributions of this report are as follows:

- We introduce UVSRNet, a novel unrolled VSR architecture that integrates model-based optimization with learning-based priors.
- We demonstrate how UVSRNet utilizes subpixel motion and autoregressive information from prior reconstructions to enhance super-resolution quality.
- We compare UVSRNet with an unrolled single image super-resolution method and a CNN-based VSR method, demonstrating UVSRNet's ability to recover small details in simulated videos.

Through these contributions, this research advances the state of VSR for distant-object recognition, providing a pathway toward interpretable, efficient, and effective solutions for critical national security applications.

2. RELATED WORK

In this section, we briefly outline the research in VSR over the last 40 years. Then, we provide an overview of unrolled methods with a focus on those applied to super-resolution. A more thorough review on VSR and motion estimation can be found in our literature review report [2], which is available upon request.

2.1 VSR METHODS

Researchers began exploring the problem of VSR as early as the late 1980s [3, 4, 5, 6, 7]. Most methods proposed prior to 2015 relied on iterative optimization frameworks that fused multiple LR frames to exploit sub-pixel information. Frame registration was typically achieved either through explicit motion estimation using optical flow [8] or by jointly estimating motion within the iterative optimization process [9]. Since super-resolution is well-known to be inherently ill-posed [6], regularization is essential to constrain the solution space by imposing prior assumptions on the HR output. Many different types of regularization have been proposed for VSR, including bilateral total variation [10] and non-local similarity [11]. While these methods made substantial progress in VSR, they tended to suffer from high computational cost due to the large number of iterations and relied on high-fidelity motion estimation, which degraded performance in the presence of complex motion, occlusions, or noise.

To reduce reliance on handcrafted priors and improve flexibility, researchers explored ways to incorporate data-driven components into iterative frameworks. In an attempt to reap the benefits of both iterative optimization methods and end-to-end deep learning methods, Venkatakrishnan et al. proposed the concept of PnP priors in 2012, which uses deep learning-based denoisers in place of explicit regularization within an iterative optimization method [12]. Since then, several researchers proposed the use of PnP priors within iterative optimization approaches for VSR [13, 14, 15, 16]. These approaches offer flexibility by allowing state-of-the-art image denoisers to act as implicit priors, while preserving the interpretability of optimization-based formulations. However, their reliance on iterative processing leads to high computational cost and long inference times compared to end-to-end networks. Furthermore, the denoiser is often trained independently of the reconstruction task, which can limit performance compared to fully learned solutions.

In light of the success of deep learning models for computer vision in the mid-2010s, researchers began designing end-to-end CNNs for VSR. VSRNet [17], proposed in 2016, was the first CNN specifically developed for VSR. VSRNet is built upon SRCNN [18], the pioneering SISR CNN composed of three convolutional layers. To exploit temporal information, VSRNet aligns three neighboring frames using optical flow estimation and frame warping, and then concatenates them between the first and second convolutional layers. In 2017, VESPCN improved upon VSRNet by adopting a more advanced SISR backbone—ESPCN [19]—and introducing a learnable spatial transformer module for joint motion estimation [20].

As research progressed, it became evident that motion estimation played a critical role in VSR performance. To overcome the limitations of explicit motion estimation, subsequent methods incorporated motion estimation directly into the network or eliminated it entirely. DUF [21] exemplified this trend by generating dynamic, spatially-varying upsampling filters based on input frames. Similarly, TOFlow [22] and SOF-VSR [23] embedded learnable motion estimation modules, while more advanced architectures such as RBPN [24] and EDVR [25] employed recurrent structures and deformable convolutions to achieve robust alignment and feature fusion. These developments led to substantial gains over traditional optical-flow-based approaches. BasicVSR [26] further simplified the pipeline by adopting a bidirectional recurrent architecture with feature propagation across frames, while its successor BasicVSR++ [27] introduced second-order propagation and flow-guided deformable alignment for improved temporal consistency. CNN-based methods generally achieve strong performance with reasonable computational cost, but are limited in capturing long-range temporal dependencies, and often require accurate motion estimation to handle complex motion and occlusions.

The introduction of transformers by researchers at Google in 2017 [28] revolutionized many areas of deep learning, including computer vision. In VSR, transformers were first applied to capture spatial dependencies through the SwinIR framework, which was later extended to video restoration tasks [29]. Subsequent work aimed to explicitly model temporal dependencies across frames; TTVSR [30] proposed a transformer-based architecture for temporal feature fusion, while RVRT [31] integrated recurrent propagation with transformer blocks to efficiently capture long-range spatio-temporal information. More recently, RealViFormer [32] further advanced spatio-temporal modeling by leveraging recurrent transformer blocks combined with channel attention, achieving strong performance on real-world VSR benchmarks. Transformer-based methods offer superior modeling of long-range spatial and temporal dependencies, and can better handle complex motion. However, they tend to be computationally intensive, require large amounts of training data, and may be prone to overfitting in low-data scenarios.

In recent years, denoising diffusion models have been increasingly applied to VSR. MGLD-VSR [33] introduced a motion-guided latent diffusion framework to capture both fine-grained details and global temporal consistency, demonstrating the potential of generative models for photorealistic video restoration. Later, Upscale-A-Video [34] applied latent diffusion for high-quality video upscaling, while improving temporal consistency across frames. Most recently, DiffVSR [35] leveraged probabilistic diffusion modeling to generate high-frequency details in real-world videos. Despite their impressive reconstruction quality, diffusion-based methods remain computationally intensive, often requiring hundreds of inference steps, and may produce temporal inconsistencies if motion or temporal regularization is insufficient.

2.2 UNROLLED METHODS

Deep unrolling, also known as deep unfolding, is a method of reformulating an iterative algorithm as a fixed-depth neural network with learnable parameters. It was originally applied to the iterative shrinkage-thresholding algorithm (ISTA) for sparse coding in 2010 by Gregor and LeCun [36]. In 2014, Hershey et al. formalized deep unrolling as a general framework for mapping iterative algorithms to end-to-end learnable architectures [37].

Building on these foundations, early work extended unrolling beyond sparse coding to a wide range of iterative procedures [38]. Notable examples include ADMM-Net [39], which reformulated the alternating direction method of multipliers (ADMM) for image reconstruction with learnable convolutional updates, as well as primal-dual hybrid gradient methods and approximate message passing (AMP) [40, 41]. These approaches demonstrated that learned update rules could accelerate convergence while maintaining the interpretability of classical solvers, paving the way for practical applications in inverse problems such as compressed sensing, deconvolution, and MRI reconstruction.

Concurrently, PnP methods emerged as an alternative route to integrating deep learning into iterative reconstruction pipelines (as introduced in Section 2.1). Unlike fixed unrolled networks, PnP methods decouple the data-consistency update from the prior, allowing advanced denoisers such as DnCNN [42] or DRUNet [1] to be incorporated seamlessly. More recent works have combined unrolled optimization backbones with PnP-inspired priors, creating hybrid architectures that retain algorithmic interpretability while leveraging strong data-driven regularization [43, 44, 45].

Several researchers have used unrolled PnP frameworks for image super-resolution. For example, USRNet [1] and ISTAR [46] iteratively refine HR reconstructions by integrating advanced denoisers at each stage, while maintaining algorithmic transparency. In USRNet, the classical maximum a posteriori (MAP) formulation of the super-resolution problem is unfolded into a fixed number of iterations, where each alternates between a data-fidelity update and a learned CNN-based denoiser, effectively combining principled model-based optimization with data-driven regularization. Similarly, ISTAR extends ISTA into an unrolled framework for super-resolution, embedding learnable proximal mappings that act as implicit priors. These approaches exemplify a broader trend in unrolled PnP methods: instead of treating the denoiser as a generic black-box, the network structure and update rules are carefully aligned with the underlying optimization algorithm. This adaptation leads to improved convergence, reduced computational overhead, and enhanced reconstruction fidelity compared with purely black-box or end-to-end CNN approaches.

In 2021, Chiche et al. proposed an unrolled framework for VSR (UVSR) [47]. UVSR consists of an unrolled gradient descent network with two learnable modules: one which estimates the LR optical flow map from the previous frame to the current frame, and one that processes the current frame using the previous super-resolved frame, the previous LR frame, and the current LR frame. However, both modules operate entirely in the LR domain, which may limit reconstruction quality. Rather than incorporating learnable layers to explicitly map features between low- and HR spaces, the framework relies on fixed space-to-depth and depth-to-space operators. Furthermore, the prior term architecture is built on a a super-resolution backbone, although in this setting the prior module is not employed to super-resolve a frame. Unlike PnP priors based on additive white gaussian noise (AWGN) denoisers, which are supported by a well-established theoretical foundation, this design choice lacks a rigorous justification. Finally, UVSR assumes that the adjoint of the motion operator is equivalent to its inverse. This identity only holds for strictly integer translational motion, not in the more general sub-pixel or non-rigid motion cases typical of real-world video sequences.

Motivated by these limitations, we propose an unrolled VSR network—UVSRNet—grounded in PnP theory and inspired by the design of USRNet [1]. By explicitly formulating VSR within a principled MAP optimization framework, our method alternates between data-consistency updates and a denoiser prior that is both theoretically justified and empirically effective. Unlike UVSR, our approach leverages a denoiser that operates in the HR domain, avoiding the loss of fidelity inherent in LR-only updates. By using algorithm unrolling, we retain the interpretability and convergence guarantees of iterative optimization, while enhancing reconstruction fidelity and temporal consistency across frames via end-to-end training. This establishes our framework as a principled and flexible alternative that unites the strengths of PnP priors with the efficiency of unrolled architectures for VSR.

3. METHOD

In this section, we present our proposed unrolled VSR framework, UVSRNet. We begin by outlining the assumed forward model and defining the autoregressive MAP estimate. Next, we describe our proposed autoregressive unrolled optimization method for computing the MAP estimate. Finally, we provide a detailed description of the deep unrolled network architecture and the end-to-end training procedure.

3.1 FORWARD MODEL

Let $x_i \in \mathbb{R}^{N_p}$ denote the rasterized ith frame of the unknown HR video to be recovered. We assume that the jth frame of the observed LR video is given by

$$y_j = A_{i\to j}x_i + \epsilon_j, \tag{1}$$

where $y_j \in \mathbb{R}^{N_p/L^2}$ is the rasterized jth LR frame, $A_{i\to j}$ is an operator that registers the ith HR frame to the jth LR frame and reduces its resolution by a factor of L, and $\epsilon_j \sim \mathcal{N}(0, \sigma^2 I)$ is independent and identically distributed (i.i.d.) AWGN.

We model $A_{i\to j}$ as the composition of a motion operator $E_{i\to j}$, a blurring operator B, and a downsampling operator D, i.e. $A_{i\to j}x_i = DBE_{i\to j}x_i$. Specifically:

- $E_{i\to j}$ registers the ith frame to the jth frame, assuming 2D affine motion (translation, rotation, and scaling),
- B convolves the image with a Gaussian kernel, with standard deviation and kernel size determined by the super-resolution factor L,
- D performs subsampling by selecting every Lth pixel along both spatial dimensions.

Together, these assumptions define a physically motivated forward model that captures the effects of blur, downsampling, and motion, providing a principled basis for recovering HR video frames from their LR observations.

3.2 AUTOREGRESSIVE UNROLLED OPTIMIZATION

Our method estimates the ith HR frame given a set of LR frames

$$Y_i := \{y_j\}_{i-w}^{i+w} \tag{2}$$

and a set of previously super-resolved frames

$$\hat{x}_{<i} := \{\hat{x}_{i-k}\}_1^p, \tag{3}$$

where w and p are user-defined parameters specifying the number of neighboring LR frames and previously recovered HR frames, respectively.

We define the data-fidelity term as the average negative log-likelihood across all frames in Y_i:

$$f(x_i; Y_i) := \frac{1}{2w+1}\sum_{j=i-w}^{i+w}\frac{1}{2\sigma^2}\|y_j - A_{i\to j}x_i\|^2 \tag{4}$$

The autoregressive MAP estimate is then defined as

$$\hat{x}_i = \arg\min_{x_i} \left\{ f(x_i; Y_i) + \lambda g(x_i; \hat{x}_{<i}) \right\}. \tag{5}$$

where $g(x_i; \hat{x}_{<i})$ is a prior term that regularizes x_i using information from previously super-resolved frames and $\lambda > 0$ is a weighting parameter that controls the amount of regularization.

To obtain an unrolled optimization method for (5), we use HQS, a quadratic penalty method with alternating minimization. Namely, we decouple the regularizer using an auxiliary variable z_i,

$$\hat{x}_i = \arg\min_{x_i} \left\{ f(x_i; Y_i) + \lambda g(z_i; \hat{x}_{<i}) \right\} \text{ such that } x_i = z_i. \tag{6}$$

Then, instead of enforcing this constraint directly, we use a quadratic penalty term with tunable weight parameter γ

$$\hat{x}_i = \arg\min_{x_i} \left\{ f(x_i; Y_i) + \lambda g(z_i; \hat{x}_{<i}) + \frac{1}{2\gamma^2} \|x_i - z_i\|^2 \right\}. \tag{7}$$

Solving this using alternating minimization results in an iterative algorithm that alternates between a data-fitting and prior sub-problem, i.e.

$$\hat{z}_i^{(k)} = \arg\min_{z_i} \left\{ f(x_i; Y_i) + \frac{1}{2\gamma^2} \|x_i - z_i\|^2 \right\} \tag{8}$$

$$\hat{x}_i^{(k)} = \arg\min_{x_i} \left\{ g(z_i; \hat{x}_{<i}) + \frac{1}{2\lambda\gamma^2} \|x_i - z_i\|^2 \right\}. \tag{9}$$

The quadratic penalty weight γ is typically scheduled to decrease over iterations, allowing the algorithm to initially focus on solving each sub-problem independently and then gradually enforce consistency between x_i and z_i as the iterations progress.

The data-fitting sub-problem,

$$\hat{z}_i^{(k)} = \arg\min_{z_i} \left\{ f(x_i; Y_i) + \frac{1}{2\gamma^2} \|x_i - z_i\|^2 \right\}, \tag{10}$$

is quadratic in z_i and can be solved efficiently using conjugate gradient method (CGM). CGM is particularly well-suited in this situation because the Hessian of the objective is symmetric and positive definite, and explicitly forming it would be computationally expensive due to the large size of $A_{i \to j}$. Instead, CGM iteratively computes matrix-vector products with $A_{i \to j}^\top A_{i \to j}$, avoiding explicit matrix construction while converging to the exact solution in a finite number of steps. In practice, a small fixed number of iterations is sufficient to achieve high-quality approximations, which makes the method both memory- and computation-efficient. This iterative approach naturally integrates into the unrolled network, allowing backpropagation through the CGM steps during end-to-end training.

For the prior sub-problem,

$$\hat{x}_i^{(k)} = \arg\min_{x_i} \left\{ g(z_i; \hat{x}_{<i}) + \frac{1}{2\gamma^2} \|x_i - z_i\|^2 \right\}, \tag{11}$$

we adopt a PnP approach, where the prior subproblem is implicitly represented by a learned denoiser which removes AWGN with standard deviation $\gamma\sqrt{\lambda}$ [12]. This strategy enables the use of a powerful deep network as a prior without requiring an explicit regularization function, allowing the model to capture complex temporal dependencies from previously super-resolved frames while remaining flexible and data-driven.

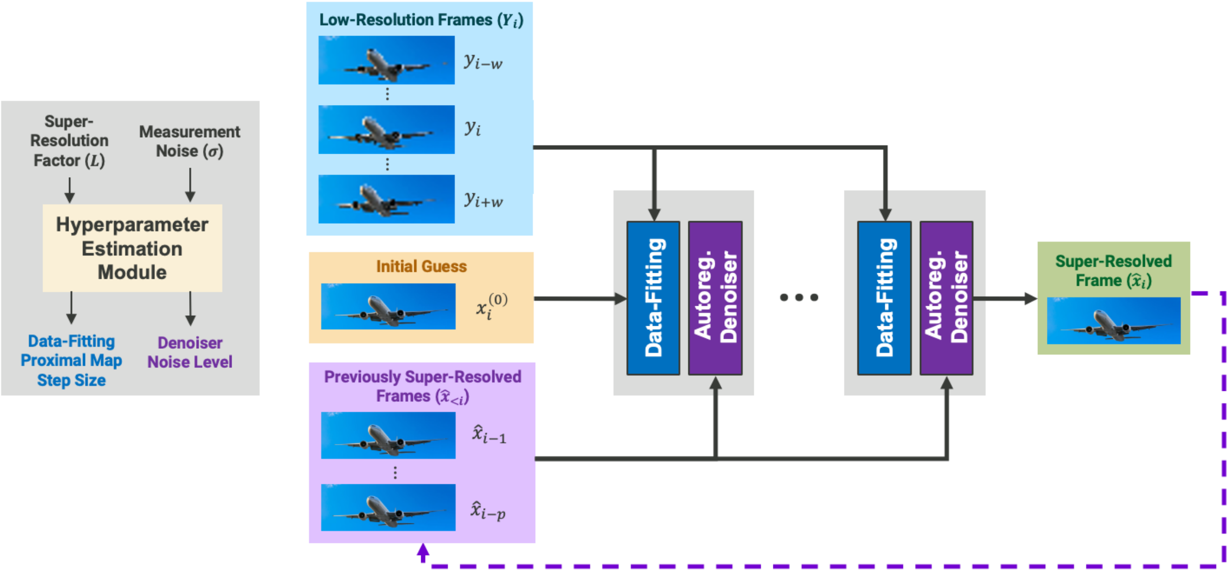

Super-Resolution Factor (L)
Measurement Noise (σ)
Hyperparameter Estimation Module
Data-Fitting Proximal Map Step Size
Denoiser Noise Level
Low-Resolution Frames (Y_i)
y_{i-w}
y_i
y_{i+w}
Initial Guess
$x_i^{(0)}$
Previously Super-Resolved Frames ($\hat{x}_{<i}$)
$\hat{x}_{i-1}$
$\hat{x}_{i-p}$
Data-Fitting
Autoreg. Denoiser
Data-Fitting
Autoreg. Denoiser
Super-Resolved Frame ($\hat{x}_i$)

Figure 1. Overall pipeline of UVSRNet for super-resolving frame i. The network alternates between a data-fitting module, which enforces consistency with the observed LR frames, and an autoregressive denoiser, which serves as a learned prior. A hyperparameter estimation module (from USRNet [1]) adaptively predicts the iteration-specific hyperparameters that balance these two components.

3.3 DEEP UNROLLED NETWORK

With this unrolled optimization algorithm as our foundation, we construct the proposed unrolled VSR network (UVSRNet). UVSRNet iteratively alternates between a data-fitting module, which enforces consistency with the observed LR frames by solving (10), and an autoregressive denoiser, which serves as a learned prior in place of (11). Since both modules depend on iteration-specific hyperparameters, we incorporate the hyperparameter estimation module from USRNet [1], which adaptively predicts the optimal values at each step. The overall pipeline of UVSRNet for super-resolving frame i is summarized in Figure 1.

3.3.1 DATA-FITTING MODULE

The data-fitting module addresses the subproblem in (10), enforcing consistency between the current estimate and the observed LR frames through the forward model $A_{i\to j}$. At the same time, it prevents large deviations from the previous estimate, where the degree of "closeness" is controlled by the step size parameter predicted by the hyperparameter module. The subproblem is solved efficiently using 5 iterations of the conjugate gradient method (with early stopping if the tolerance 10^{-5} is met). Importantly, this module contains no trainable parameters, making it highly flexible and generalizable across different forward models.

3.3.2 AUTOREGRESSIVE PRIOR MODULE

The prior module serves as a learned regularizer, producing a cleaner HR estimate x_k from the intermediate solution z_k. In our framework, we employ an autoregressive denoiser that leverages both spatial and temporal information. Specifically, the denoiser takes as input the concatenation of z_k, a noise-level map corresponding to output of the hyperparameter estimation module output, and neighboring frames from the sequence, thereby exploiting temporal redundancy to guide

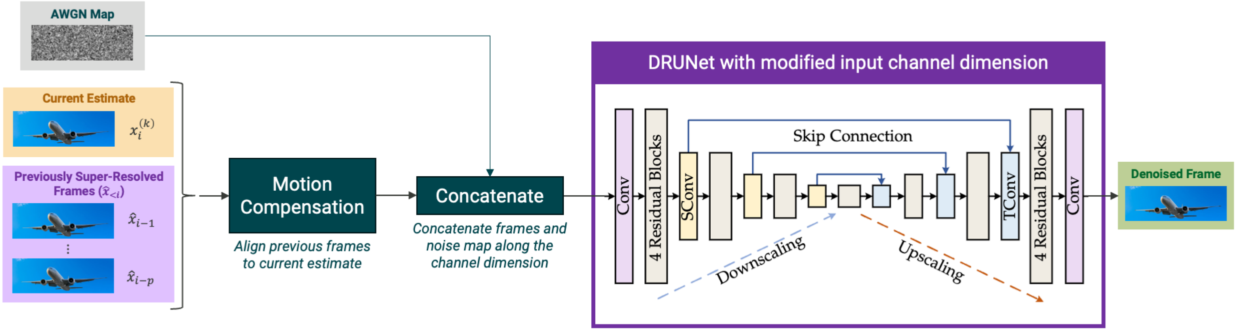


Figure 2. Architecture of the autoregressive prior module. At each iteration, the denoiser refines the intermediate solution z_k by processing the concatenation of z_k, a noise-level map, and registered neighboring frames. DRUNet [43] serves as the backbone, but we extend the input layer to handle multiple frames for exploiting temporal correlations in VSR.

the reconstruction. An overview of the autoregressive denoiser pipeline and architecture is shown in Figure 2).

As the backbone of this denoiser, we adopt the DRUNet architecture [43], a state-of-the-art restoration network that combines U-Net's multi-scale feature aggregation with residual connections for stable training and strong representational power. DRUNet naturally supports the injection of a noise map as an additional input channel, allowing a single model to adapt across varying noise levels without retraining.

Architecturally, DRUNet follows a four-scale U-Net structure with residual blocks embedded in both downsampling and upsampling paths. The number of channels increases across scales (64, 128, 256, and 512), with 2 × 2 strided convolutions for downsampling and 2 × 2 transposed convolutions for upsampling. Each residual block consists of two 3 × 3 convolutions with ReLU activations and an identity skip connection, enhancing both training stability and representational depth. In our implementation, the autoregressive design extends DRUNet by concatenating multiple input frames along the channel dimension, enabling the network to capture temporal correlations while remaining lightweight and modular. This design is particularly well-suited for VSR, as it promotes temporal consistency while reducing artifacts that arise in frame-wise processing.

3.3.3 HYPERPARAMETER ESTIMATION MODULE

We adopt the hyperparameter estimation module proposed in USRNet [1], which serves as a controller to balance the outputs of the data-fitting and prior modules. Specifically, it predicts the step size for the data-fitting update and the noise level for the denoiser prior, both of which vary across iterations. Rather than fixing these parameters, the module learns to adapt them based on two critical factors that govern the ill-posedness of the problem: the scale factor L and the measurement noise level σ. The module is implemented as a lightweight fully-connected network with three layers (64 hidden units each), using ReLU activations for the first two layers and Softplus in the output layer (with an added 10^{-6} offset to enforce positivity and numerical stability). By dynamically adapting the hyperparameters to the imaging conditions, this module balances data fidelity and prior regularization across iterations while eliminating the need for manual parameter tuning.

3.4 END-TO-END TRAINING

We employ end-to-end training to jointly optimize the trainable parameters of the autoregressive prior module and the hyperparameter estimation module. This section describes the training data, loss function, and implementation details.

For training, we generate 100 synthetic videos of size 256 × 256 with 25 frames each, derived from the REDS training set [48]. Each HR video is created by selecting the 15th frame of a REDS sequence and applying a fixed 2 pixel translation upward and leftward across 25 frames, producing temporally consistent motion. The corresponding LR videos are obtained by applying a 9 × 9 Gaussian blur with $\sigma = 1.5$, subsampling by a factor of 4, and adding AWGN with $\sigma = 0.01$. This controlled design ensures that the inter-frame motion is exactly known, removing the need for motion estimation and avoiding mis-registration artifacts. Such a setup allowed us to focus on refining the core method without confounding factors. In future work, we will extend the training pipeline by incorporating motion estimation for real video data and expanding the scale of the training set.

The loss for a single video is computed as the average L1 error over its 25 frames. Following USRNet, we apply supervision only to the final reconstructed video rather than intermediate outputs. Optimization is performed using the Adam solver [49] with an initial learning rate of 1×10^{-4}, reduced by a factor of two if the validation SSIM does not improve for 10 epochs. Training is conducted for 50 epochs using PyTorch on a single NVIDIA H100 GPU with 80 GB of memory. End-to-end training of one model for 50 epochs requires approximately 12 hours.

4. EXPERIMENTAL RESULTS

In this section, we evaluate the proposed UVSRNet on synthetic test data designed to provide controlled and interpretable comparisons. The test set consists of 15 videos of size 256 × 256 with 25 frames each, generated from the REDS dataset [48] using the same procedure as described in Section 3.4. This setup ensures known inter-frame motion, enabling a focused study of the network's behavior without the confounding effects of motion estimation errors. We assess reconstruction quality using both qualitative visualizations and quantitative metrics (peak signal-to-noise ratio (PSNR) and structural similarity index measure (SSIM)). The residuals are calculated as the mean absolute error between the reconstructed frame and the ground truth frame over the three color channels.

Our experiments are organized to systematically analyze the contributions of different design choices in UVSRNet. First, we investigate the impact of the number of PnP iterations, which controls the degree of unrolling. Next, we examine the role of the temporal window size, i.e. the number of LR frames used, and the effect of conditioning on previously super-resolved frames in the autoregressive prior. We then analyze the complementary behavior of the data-fidelity and denoiser modules, as well as the behavior of the hyperparameter estimation module across iterations. Finally, we compare UVSRNet against two baselines, the single-image USRNet [1] and the video-based RealBasicVSR [50], to highlight the strengths and limitations of our approach.

4.1 IMPACT OF NUMBER OF PNP ITERATIONS

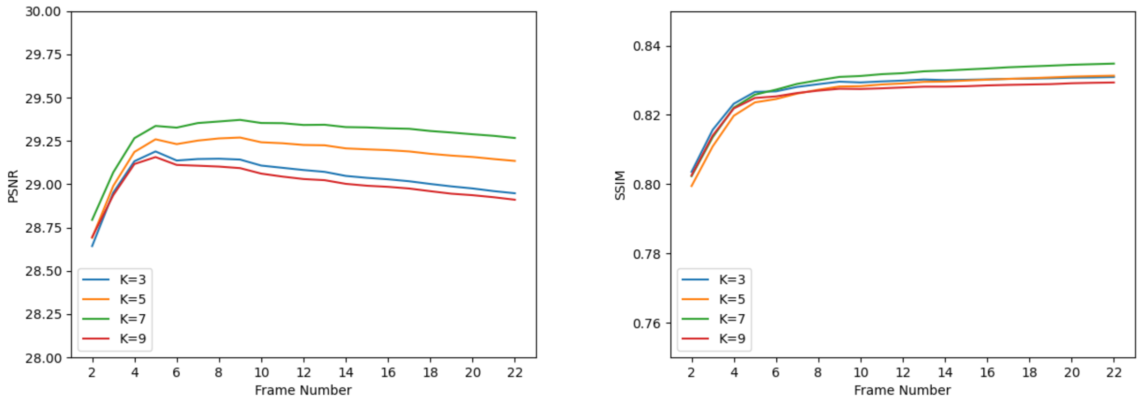
When designing unrolled networks, the number of iterations K is an important trade-off parameter. A larger number of iterations provides more opportunities for alternating between the data-fidelity and prior modules, potentially improving reconstruction quality. However, increasing K also raises both training and inference costs, and past a certain point, additional iterations may yield only marginal benefits. To identify a reasonable balance between performance and efficiency, we evaluate our method with $K = 3, 5, 7$, and 9 PnP iterations. For this experiment, the other parameters are fixed as $w = 2$ and $p = 2$.

Figure 3 shows the average PSNR and SSIM values across 15 test videos for varying numbers of PnP iterations. Reconstruction quality improves slightly as K increases, with $K = 7$ achieving the highest average performance. However, the gains over $K = 3$ and $K = 5$ are modest, and $K = 9$ provides no additional benefit. This suggests diminishing returns when using more than a few iterations.

Figure 4 provides a qualitative comparison of one cropped frame from a test video. Consistent with the quantitative results, the reconstructions are visually very similar across different values of K. Based on this observation, we adopt $K = 3$ in practice, as it offers nearly the same visual and quantitative performance while substantially reducing both training and inference time.

4.2 IMPACT OF NUMBER OF LOW-RESOLUTION FRAMES

When designing a VSR method, a natural question is how many neighboring LR frames should be incorporated into the reconstruction. Using additional frames provides more spatial–temporal information, particularly in the form of subpixel motion information that is absent when relying on a single frame. However, using too many frames may lead to redundancy, increased computational cost, and potential error accumulation from imperfect motion alignment. To better understand



(a) Average PSNR across all test videos **(b) Average SSIM across all test videos**

Figure 3. Quantitative evaluation of the impact of the number of PnP iterations K on reconstruction quality. Results are averaged over 15 simulated test videos with 25 frames each. Using $K = 7$ achieves the best performance, though differences across $K = 3, 5, 7, 9$ are relatively small. Note the limited y-axis ranges (PSNR: 28–30 dB, SSIM: 0.75–0.85).

this trade-off, we investigate the impact of varying the temporal window size w, where $2w + 1$ denotes the number of LR frames used in each reconstruction step. For this experiment, the other parameters are fixed as $K = 3$ and $p = 2$.

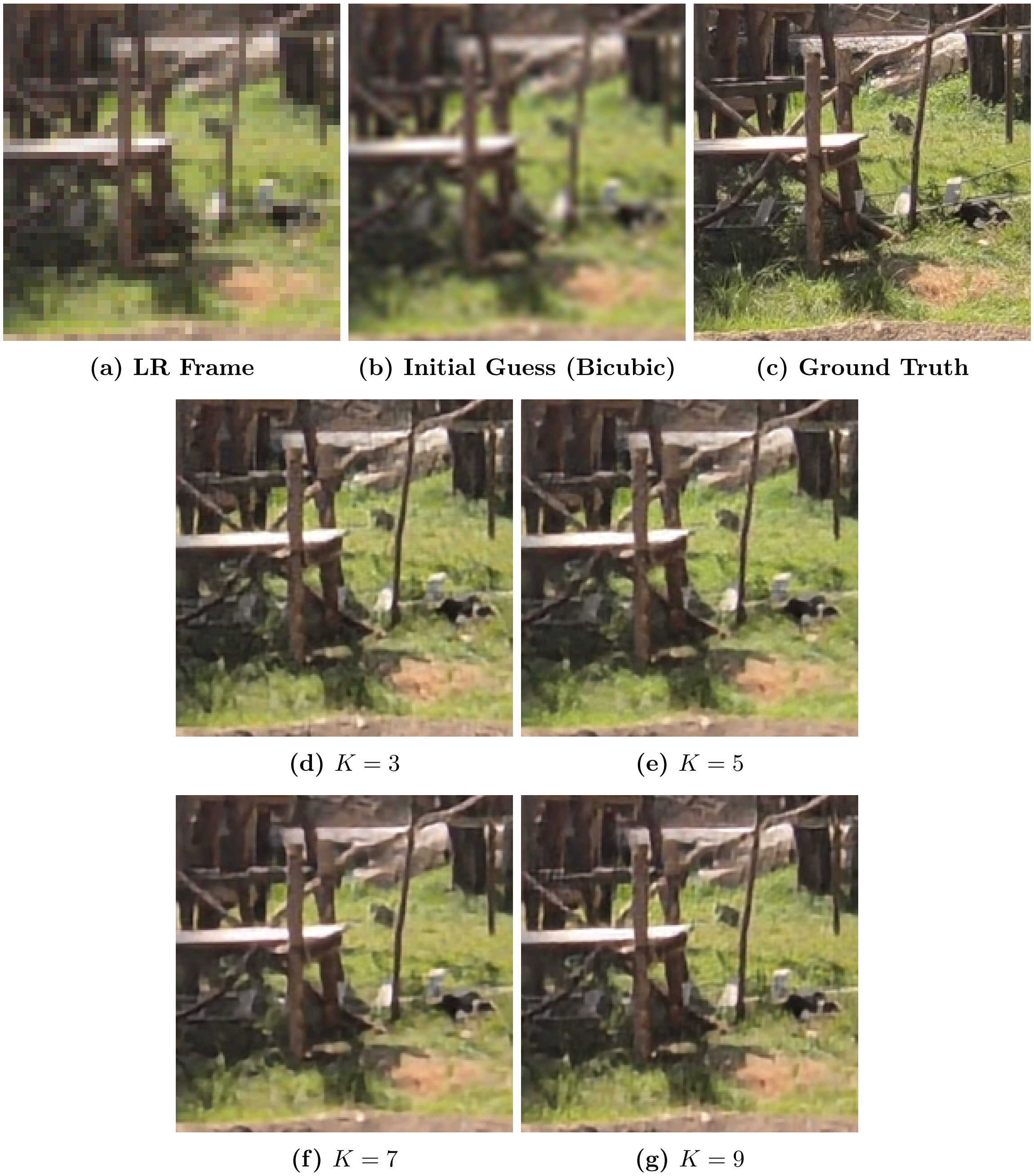
Figure 5 reports the average PSNR and SSIM across 15 test videos for varying number of LR frames. We observe that incorporating more than 1 LR frame ($w > 0$) consistently improves performance compared to the single-frame baseline ($w = 0$), confirming the value of leveraging temporal cues and subpixel motion. While $w = 2$ yields the best results, the primary performance gain arises from moving beyond the single-frame case, with only incremental improvements when adding more than one neighboring frame.

Figure 6 presents qualitative examples. While differences between $w = 1$ and $w = 2$ are subtle, both clearly outperform the single-frame case ($w = 0$), showing sharper textures and reduced residuals. These results highlight that the key benefit comes from exploiting subpixel motion through multiple frames, with $w = 2$ providing the strongest overall performance without requiring a larger temporal window.

4.3 IMPACT OF NUMBER OF PREVIOUS FRAMES

A key design choice in our autoregressive prior is the number of previously super-resolved frames used, p. Intuitively, incorporating past frames should provide useful temporal information that can improve consistency across frames. However, conditioning on too many frames may also introduce error propagation, where artifacts from earlier reconstructions accumulate and degrade performance. To better understand this trade-off, we evaluate the effect of varying p on both quantitative metrics and qualitative reconstructions. For this experiment, the other parameters are fixed as $K = 3$ and $w = 2$.

Figure 7 shows the average PSNR and SSIM values using $p = 0, 1,$ and 2. Using one previously super-resolved frame ($p = 1$) yields the highest reconstruction quality, although the margin over



(a) LR Frame

(b) Initial Guess (Bicubic)

(c) Ground Truth

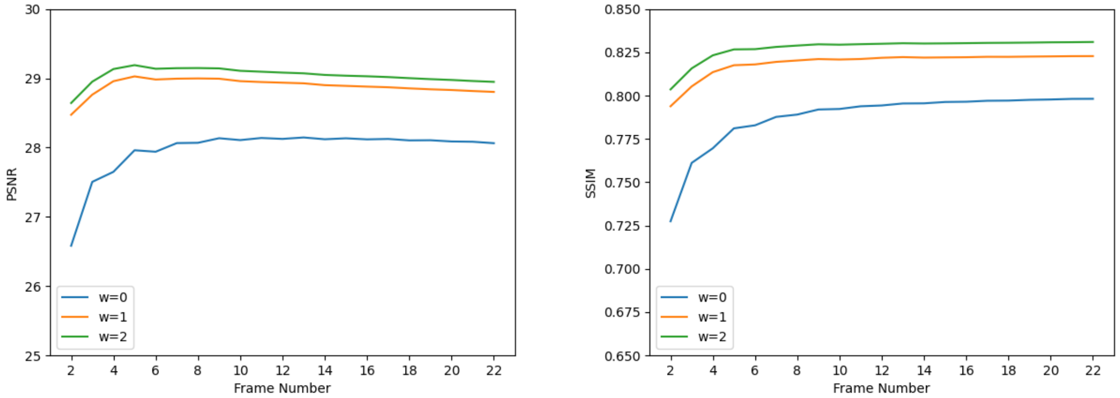
(d) $K = 3$

(e) $K = 5$

(f) $K = 7$

(g) $K = 9$

Figure 4. Qualitative comparison of reconstructions using different numbers of PnP iterations K. Results are shown for a 200×200 crop of a test video. Cropped regions show minimal visible differences across $K = 3, 5, 7, 9$, consistent with the small quantitative variations shown in Figure 3.



(a) Average PSNR across all test videos

(b) Average SSIM across all test videos

Figure 5. Quantitative evaluation of the impact of the number of LR frames w on reconstruction quality. Results are averaged over 15 simulated test videos with 25 frames each. Incorporating neighboring frames $(w > 0)$ leads to a significant improvement over the single-frame case $(w = 0)$, highlighting the benefit of exploiting subpixel motion. Among the tested settings, using 5 LR frames $(w = 2)$ achieves the best overall performance, though the difference between $w = 1$ and $w = 2$ is modest. Note the limited y-axis ranges (PSNR: 25–30 dB, SSIM: 0.65–0.85).

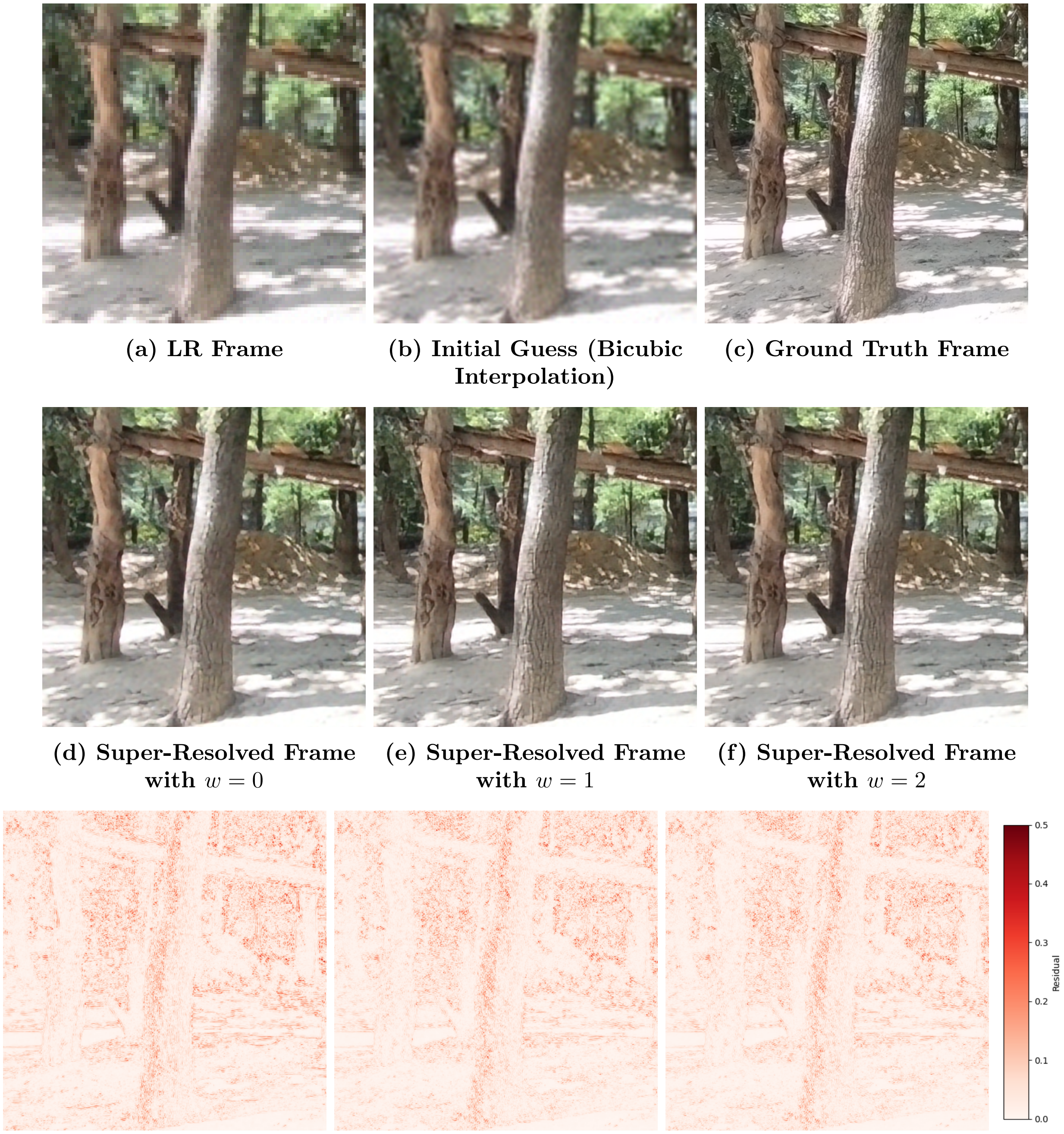
$p = 0$ and $p = 2$ is slight. This suggests that temporal information is beneficial, but relying on more than one frame offers limited gains in this controlled setting.

Figure 8 provides a qualitative comparison on a cropped region. While differences across $p = 0, 1, 2$ are subtle in the reconstructions, the residual maps reveal small variations that align with the quantitative results. Because this evaluation is performed on simulated videos with simple translational motion, the benefits of temporal conditioning are limited. We anticipate that incorporating previous frames will play a more significant role in challenging real-world scenarios involving complex motion and degradations. Based on these results, we choose to use $p = 1$ for all remaining experiments.

4.4 ANALYSIS OF DATA-FITTING AND DENOISER MODULES

In this subsection, we examine the respective contributions of the data-fitting module and the autoregressive denoiser across iterative updates. Figure 9(a) illustrates the outputs of both modules for 4× super-resolution of frame 22 from a simulated LR video, with the top row corresponding to the data-fitting module and the bottom row to the autoregressive denoiser. Figure 9(b) provides a magnified view of the cropped region indicated by the green box in the ground-truth frame of Figure 9(a).

The two modules exhibit complementary behavior: the data-fitting module primarily sharpens structural details and delineates edges, while the autoregressive denoiser suppresses blocking artifacts and noise, thereby enhancing perceptual quality. The alternating application of these modules across iterations yields reconstructions of substantially higher fidelity than those obtained from either module in isolation.



(a) LR Frame

(b) Initial Guess (Bicubic Interpolation)

(c) Ground Truth Frame

(d) Super-Resolved Frame with $w = 0$

(e) Super-Resolved Frame with $w = 1$

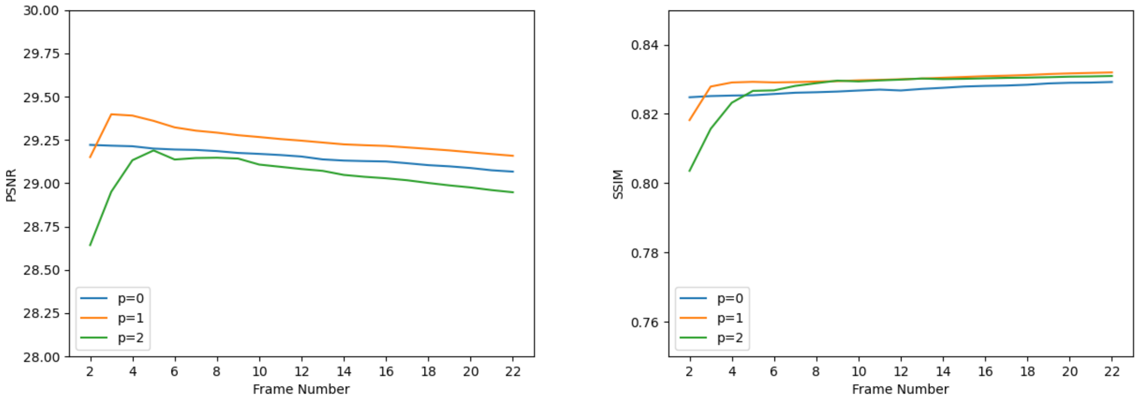
(f) Super-Resolved Frame with $w = 2$

(g) Residual for Super-Resolved Frame with $w = 0$

(h) Residual for Super-Resolved Frame with $w = 1$

(i) Residual for Super-Resolved Frame with $w = 2$

Figure 6. Qualitative comparison of reconstructions using different numbers of LR frames w. Results are shown for a 200×200 crop of a test video. Using $w > 0$ produces visibly sharper textures and reduces residual error compared to the single-frame case ($w = 0$). The difference between $w = 1$ and $w = 2$ is minor, consistent with the quantitative results in Figure 5.



(a) Average PSNR across all test videos (b) Average SSIM across all test videos

Figure 7. Quantitative evaluation of the impact of the number of previous frames p on reconstruction quality. Results are averaged over 15 simulated test videos with 25 frames each. Using $p = 1$ achieves the best performance, though differences across $p = 0, 1, 2$ are relatively small. Note the limited y-axis ranges (PSNR: 28–30 dB, SSIM: 0.75–0.85).

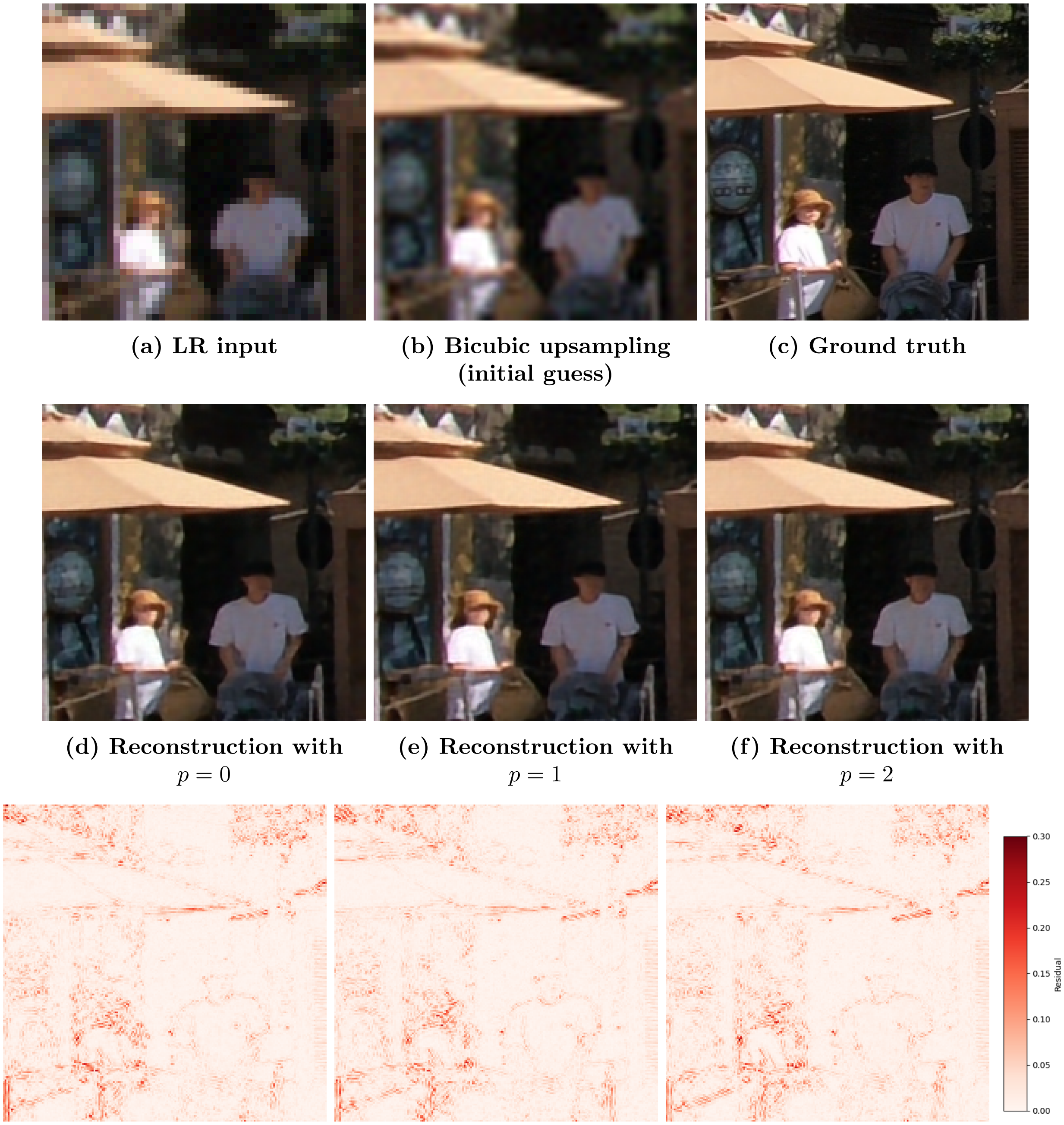
4.5 ANALYSIS OF HYPERPARAMETER ESTIMATION MODULE

Figure 10 shows the outputs of the hyperparameter module under the assumed forward model of 4× super-resolution with additive white Gaussian noise (AWGN) of standard deviation $\sigma = 0.01$. Both the data-fitting step size and the denoiser noise level decrease progressively across iterations, a behavior that mirrors the convergence properties of HQS and reflects the algorithm's transition from coarse updates to fine refinements. Notably, the data-fitting step size remains consistently larger than the denoiser noise level, underscoring a design in which data fidelity exerts stronger influence than regularization. This weighting suggests that the method prioritizes adherence to the observation model while still leveraging the autoregressive denoiser to suppress residual artifacts, thereby striking a balance between reconstruction accuracy and perceptual quality.

4.6 COMPARISON TO OTHER SUPER-RESOLUTION METHODS

In this section, we compare the proposed UVSRNet with two state-of-the-art baselines: USRNet, a single-image super-resolution method, and RealBasicVSR, a VSR approach. We use the official implementations for each of these methods, which can be found at https://github.com/ckkelvinchan/RealBasicVSR and https://github.com/cszn/KAIR. Note that neither USRNet or RealBasicVSR use the known-motion assumption that UVSRNet does. Since USRNet is a single-image super-resolution method that operates on only one LR frame, we include it primarily as a baseline. This comparison is useful because UVSRNet extends USRNet to the video setting; thus, demonstrating improvement over USRNet confirms that leveraging multi-frame information indeed enhances reconstruction quality.

Figure 11 shows the average PSNR and SSIM values over the 15 test videos for each method. UVSRNet consistently outperforms both USRNet and RealBasicVSR for all 25 frames, achieving the highest average PSNR and SSIM values. The improvement relative to USRNet highlights the benefit of incorporating temporal information. Although RealBasicVSR is also designed for



(a) LR input

(b) Bicubic upsampling (initial guess)

(c) Ground truth

(d) Reconstruction with $p = 0$

(e) Reconstruction with $p = 1$

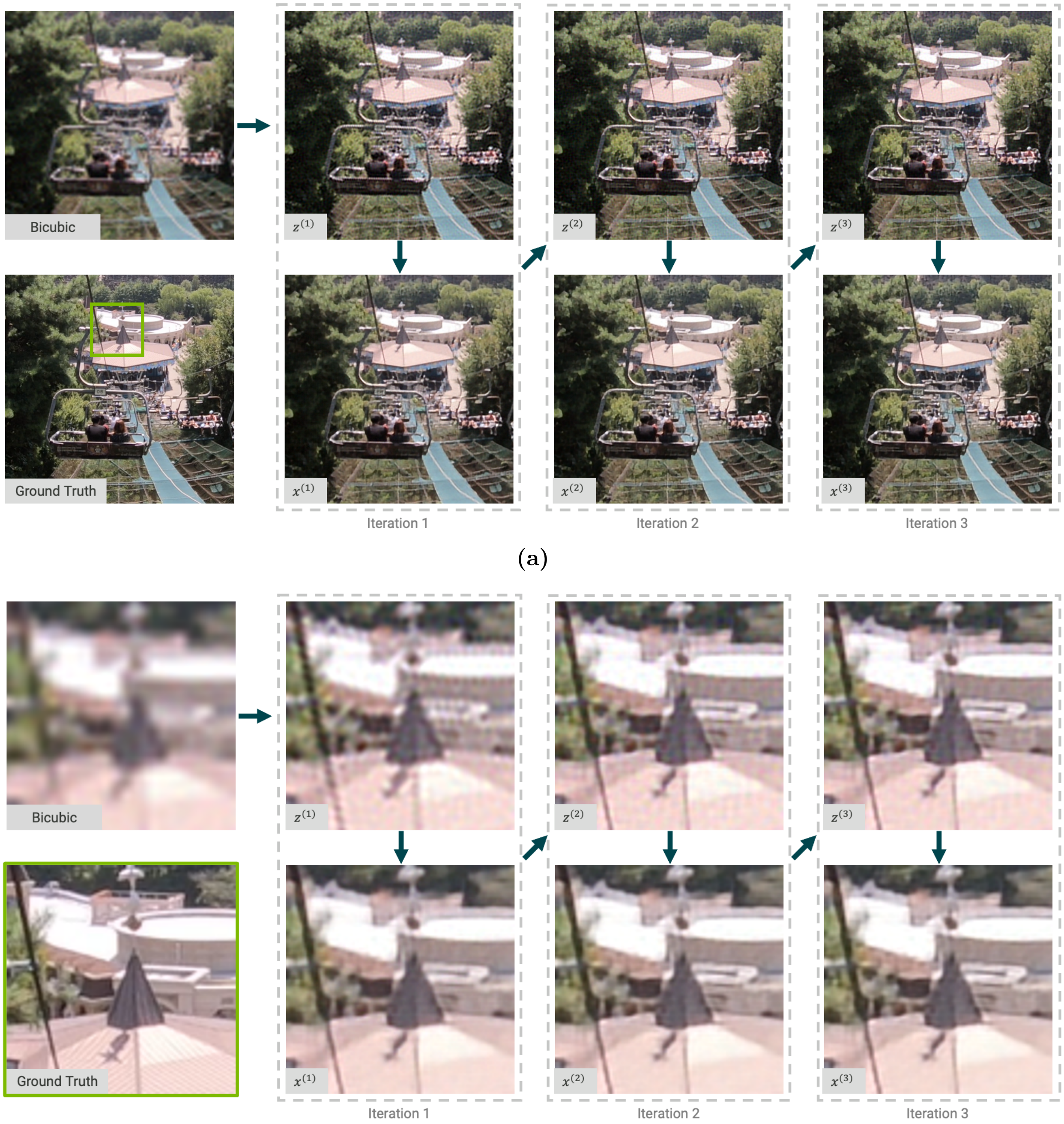
(f) Reconstruction with $p = 2$

(g) Residual for $p = 0$

(h) Residual for $p = 1$

(i) Residual for $p = 2$

Figure 8. Qualitative comparison of reconstructions using different numbers of previous frames p. Results are shown for a 200×200 crop of a test video. Differences among $p = 0, 1, 2$ are visually small, consistent with the quantitative results.



(a)

(b)

Figure 9. (a) Outputs of the data-fitting module (top row) and autoregressive denoiser (bottom row) across iterations for 4× super-resolution, with green box indicating the cropped region. (b) Magnified view of the cropped area, showing that the data-fitting module sharpens structures while the denoiser suppresses artifacts; their alternation yields higher-fidelity reconstructions.

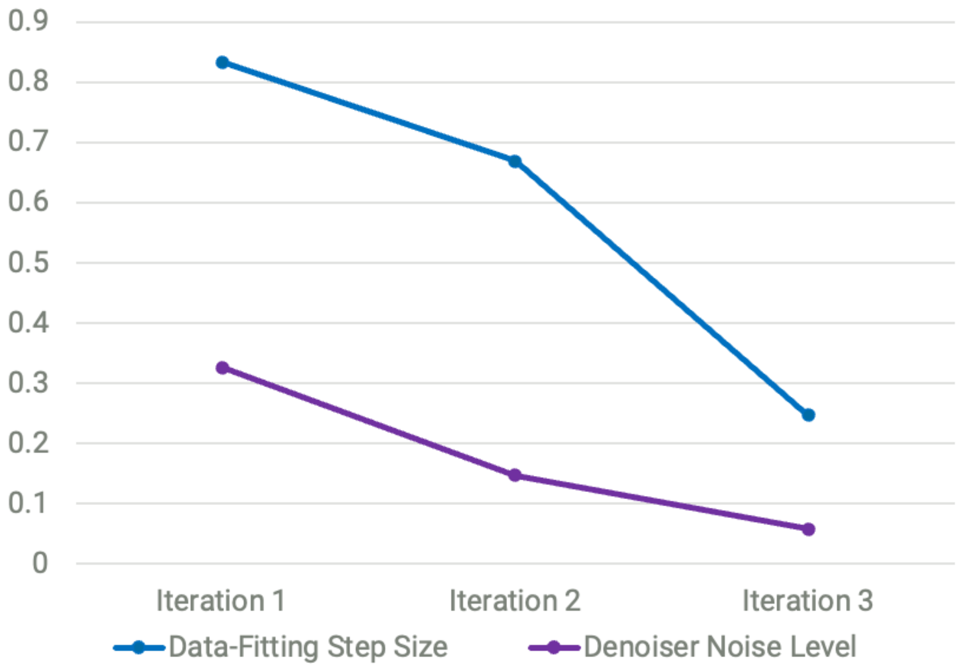
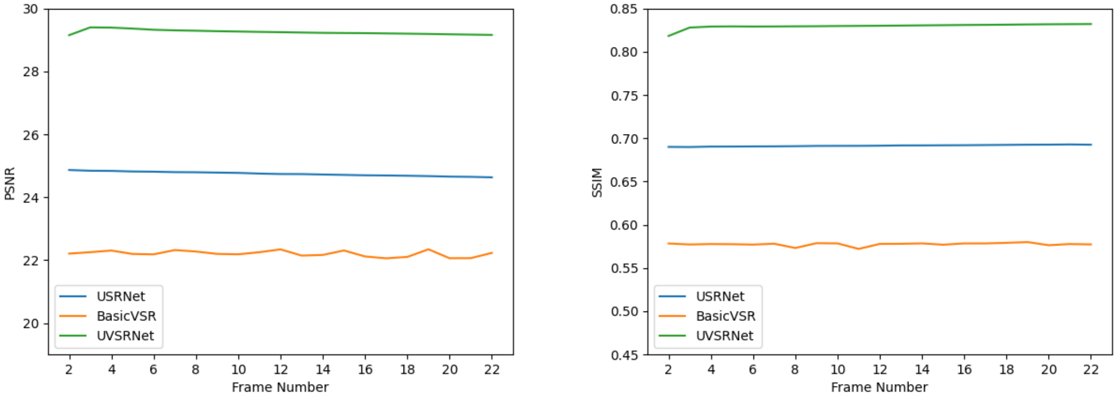


Figure 10. Output of the hyperparameter module under $4\times$ super-resolution forward model with AWGN of standard deviation $\sigma = 0.01$. Both the data-fitting step size and the denoiser noise level decrease across iterations, reflecting HQS-like convergence from coarse to fine updates. The data-fitting step size remains consistently larger, indicating stronger weighting on data fidelity relative to regularization, thereby balancing reconstruction accuracy with artifact suppression.



(a) Average PSNR over test set

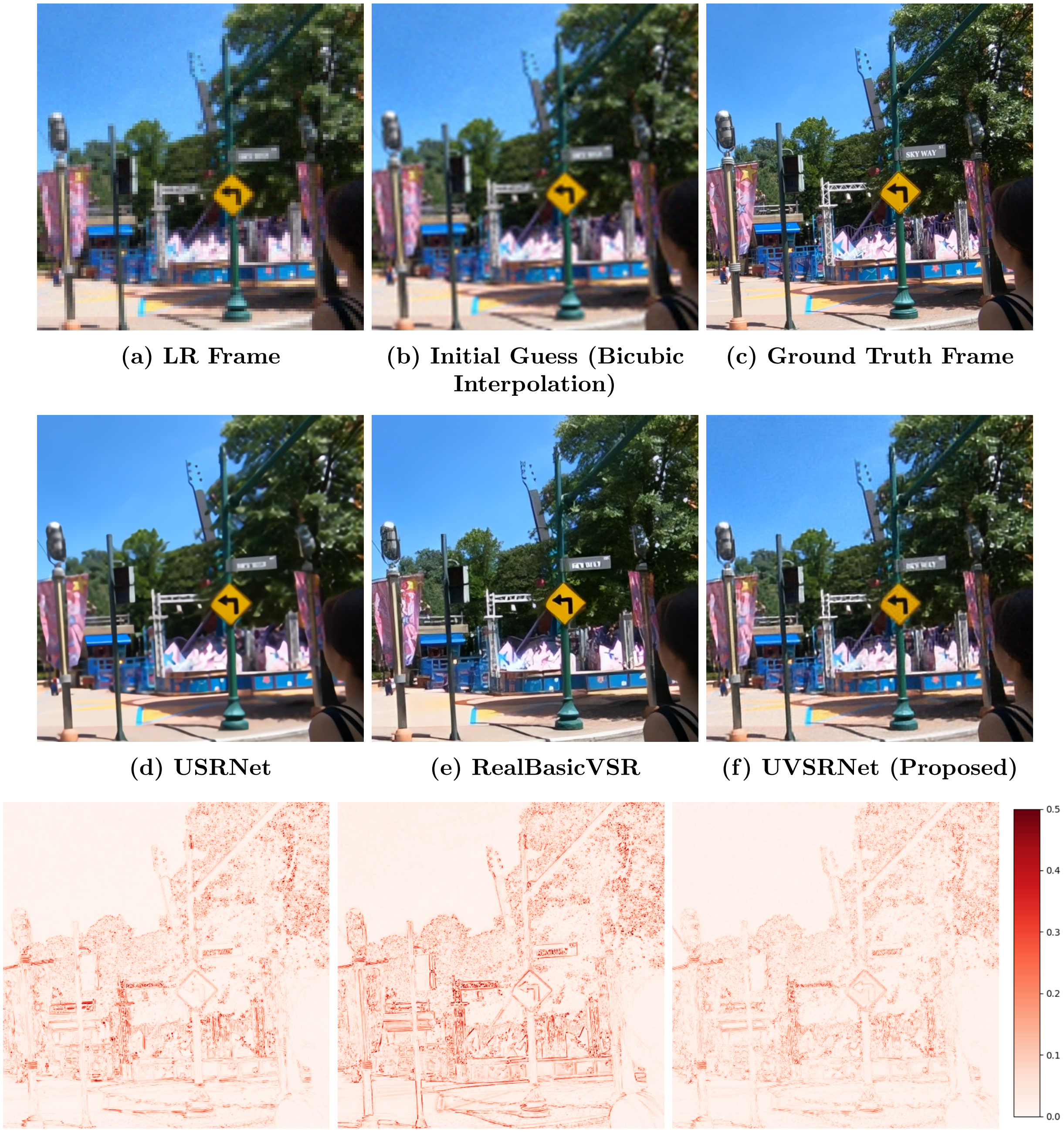
(b) Average SSIM over test set

Figure 11. Quantitative evaluation of the reconstruction quality from USRNet, RealBasicVSR, and our proposed UVSRNet. Results are averaged over 15 simulated test videos with 25 frames each. UVSRNet achieves the best performance by a large margin. Note the limited y-axis ranges (PSNR: 19.5–30 dB, SSIM: 0.45–0.85).

video inputs, UVSRNet provides a significant gain in both metrics, highlighting the effectiveness of the proposed unrolled PnP framework.

Figure 12 provides a qualitative comparison of UVSRNet with USRNet and RealBasicVSR. The top row shows (a) the LR input frame, (b) the bicubic upsampled frame, and (c) the ground-truth HR frame. The second row displays the super-resolved results from (d) USRNet, (e) RealBasicVSR, and (f) the proposed UVSRNet. The third row presents residual maps with respect to the ground truth for (g) USRNet, (h) RealBasicVSR, and (i) UVSRNet, computed as the mean absolute difference over the three color channels. Qualitatively, UVSRNet produces sharper edges and better preserves fine details than both USRNet and RealBasicVSR, while reducing the over-smoothing and artifacts observed in the baselines. The residual maps reinforce this observation: lighter and less structured residuals around object boundaries in the UVSRNet results indicate a closer match to the ground truth. Figure 13 further highlights these differences by zooming into a 150 × 150 region. In this cropped view, UVSRNet successfully reconstructs the thin vertical bars of the purple fence, which occupy only a few pixels in the LR input, whereas the other methods fail to recover this detail.

While UVSRNet retains smaller details, it is also noisier than the other two reconstructions and underperforms in capturing realistic texture in the trees compared to RealBasicVSR. This could be a result of the training set, as RealBasicVSR is trained on the entire REDS dataset while UVSRNet is currently only trained on a small subset of the REDS dataset. Future work will equalize the training sets for a fairer comparison between these two methods.



(a) LR Frame

(b) Initial Guess (Bicubic Interpolation)

(c) Ground Truth Frame

(d) USRNet

(e) RealBasicVSR

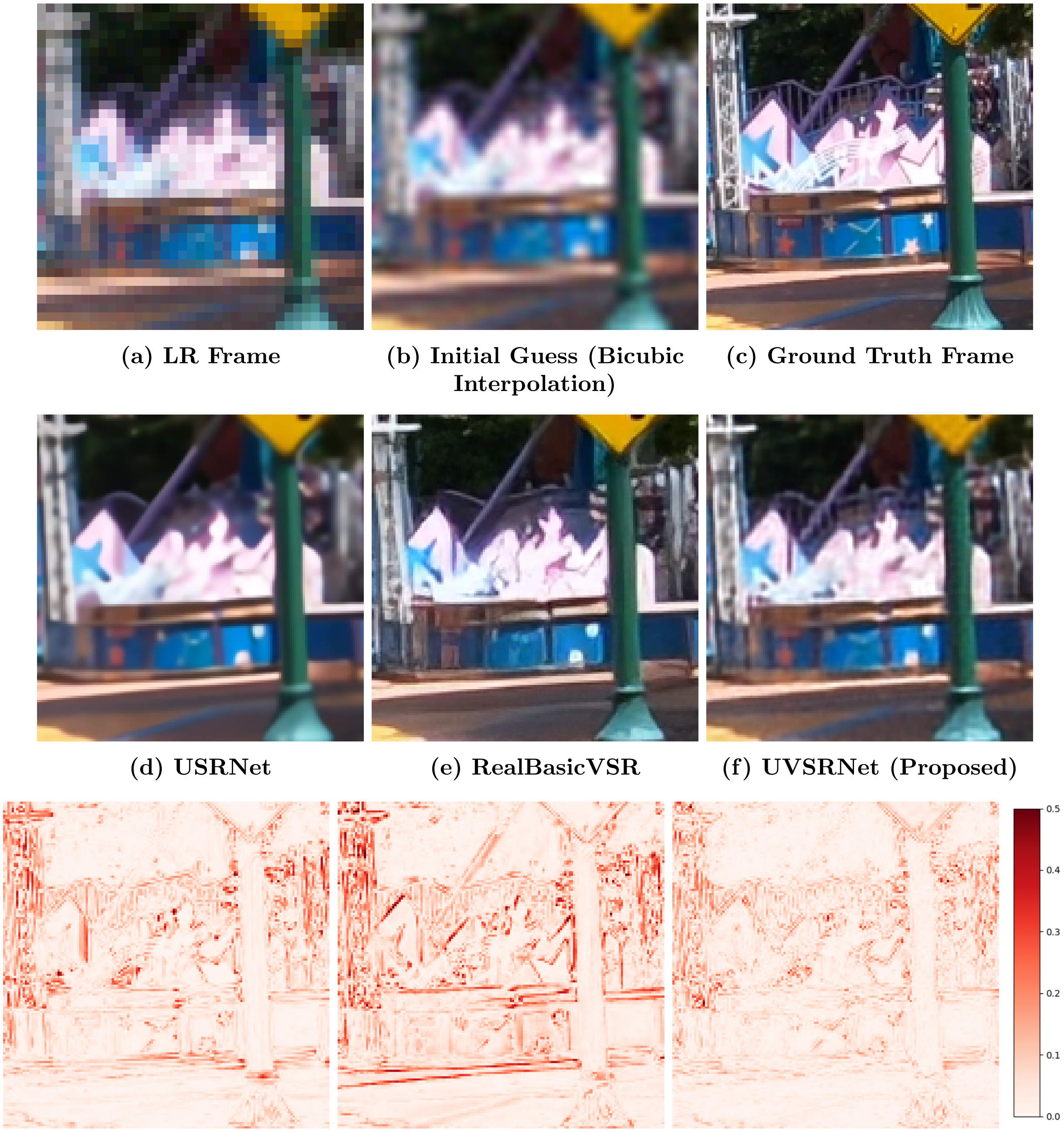
(f) UVSRNet (Proposed)

(g) USRNet Residual

(h) RealBasicVSR Residual

(i) UVSRNet Residual

Figure 12. Qualitative comparison of UVSRNet with USRNet and RealBasicVSR. The first row provides context with the LR, bicubic, and ground-truth HR frames, while the second row shows the reconstructions from the three methods. The residual maps in the third row highlight differences with respect to the ground truth. UVSRNet produces sharper edges, better preserves fine details, and yields smaller residuals compared to the baselines.



(a) LR Frame

(b) Initial Guess (Bicubic Interpolation)

(c) Ground Truth Frame

(d) USRNet

(e) RealBasicVSR

(f) UVSRNet (Proposed)

(g) USRNet Residual

(h) RealBasicVSR Residual

(i) UVSRNet Residual

Figure 13. Zoomed-in comparison on a 150×150 region from Figure 12. UVSRNet more faithfully reconstructs fine structures—such as the thin vertical bars of the purple fence—that are lost or blurred in USRNet and RealBasicVSR.

5. FUTURE WORK

Our immediate priority is to extend the current method to real video data. This requires the addition of a motion estimation module to handle sequences with unknown inter-frame motion. As a first step, we will incorporate a classical optical flow method, such as Lucas-Kanade [51], to explicitly register frames. Once this baseline is established, we aim to replace the fixed optical flow component with a learnable motion estimation module. A trainable module can adapt to complex motion patterns and mitigate artifacts introduced by explicit registration, which have been reported in prior work [20].

After enabling UVSRNet to operate reliably on real videos, we plan to refine the reconstructions through adversarial fine-tuning. Specifically, we will fine-tune the UVSRNet mdoel weights by incorporating a discriminator network, which will encourage perceptual fidelity and sharper textures, complementing the pixel-wise reconstruction loss. This approach was used to refine USRNet, yielding better visually pleasant results with smaller structures and finer textures [1].

Finally, we intend to explore a generative PnP [52] extension of UVSRNet. While the current framework focuses on MMSE estimation, the generative PnP formulation would enable sampling from the posterior distribution of HR videos. This would not only provide point estimates but also quantify uncertainty and generate multiple plausible reconstructions, opening the door to applications in simulation, video synthesis, and uncertainty-aware decision making.

6. CONCLUSION

In this work, we presented UVSRNet, a novel unrolled VSR architecture designed to address the challenges of distant-object recognition from LR video. By unrolling a PnP HQS framework into an end-to-end trainable network, UVSRNet combines the interpretability and flexibility of model-based optimization with the efficiency and reconstruction quality of learning-based methods. Our autoregressive prior module leverages temporal dependencies to recover high-frequency details, while the data-consistency module ensures fidelity to observed LR frames.

Through experiments on synthetic video data, we demonstrated that UVSRNet is able to recover fine-grained spatial detail and outperform both single-image unrolled super-resolution and a conventional CNN-based VSR approach for 4× super-resolution. These results establish UVSRNet as a promising and principled framework for VSR in operationally constrained environments.

REFERENCES

[1] Kai Zhang, Luc Van Gool, and Radu Timofte. “Deep unfolding network for image super-resolution”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 3217–3226.

[2] Haley Duba-Sullivan and Emma J. Reid. *Video Super-Resolution and Motion Estimation Literature Review*. Literature Review. Oak Ridge National Laboratory (ORNL), 2025.

[3] Shmuel Peleg, Danny Keren, and Limor Schweitzer. “Improving image resolution using subpixel motion”. In: *Pattern recognition letters* 5.3 (1987), pp. 223–226.

[4] Raymond Y Tsai. “Multiple frame image restoration and registration”. In: *Advances in Computer Vision and Image Processing* 1 (1989), pp. 1715–1989.

[5] SP Kim, Nirmal K Bose, and Hector M Valenzuela. “Recursive reconstruction of high resolution image from noisy undersampled multiframes”. In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 38.6 (1990), pp. 1013–1027.

[6] Michal Irani and Shmuel Peleg. “Improving resolution by image registration”. In: *CVGIP: Graphical models and image processing* 53.3 (1991), pp. 231–239.

[7] A Murat Tekalp, Mehmet K Ozkan, and M Ibrahim Sezan. “High-resolution image reconstruction from lower-resolution image sequences and space-varying image restoration”. In: *[Proceedings] ICASSP-92: 1992 IEEE International Conference on Acoustics, Speech, and Signal Processing*. Vol. 3. IEEE. 1992, pp. 169–172.

[8] Stefanos P Belekos, Nikolaos P Galatsanos, and Aggelos K Katsaggelos. “Maximum a posteriori video super-resolution using a new multichannel image prior”. In: *IEEE transactions on image processing* 19.6 (2010), pp. 1451–1464.

[9] Ce Liu and Deqing Sun. “On Bayesian adaptive video super resolution”. In: *IEEE transactions on pattern analysis and machine intelligence* 36.2 (2013), pp. 346–360.

[10] Sina Farsiu et al. “Fast and robust multiframe super resolution”. In: *IEEE transactions on image processing* 13.10 (2004), pp. 1327–1344.

[11] Jian Lu, HongRan Zhang, and Yi Sun. “Video super resolution based on non-local regularization and reliable motion estimation”. In: *Signal Processing: Image Communication* 29.4 (2014), pp. 514–529.

[12] Singanallur V Venkatakrishnan, Charles A Bouman, and Brendt Wohlberg. “Plug-and-play priors for model based reconstruction”. In: *2013 IEEE global conference on signal and information processing*. IEEE. 2013, pp. 945–948.

[13] Alon Brifman, Yaniv Romano, and Michael Elad. “Unified single-image and video super-resolution via denoising algorithms”. In: *IEEE Transactions on Image Processing* 28.12 (2019), pp. 6063–6076.

[14] Johnathan Mulcahy-Stanislawczyk et al. *Super Resolving Unrolled Neural Networks for Remote Sensing*. Tech. rep. Sandia National Lab. (SNL-NM), Albuquerque, NM (United States), Oct. 2024. DOI: 10.2172/2480102. URL: https://www.osti.gov/biblio/2480102.

[15] Matina Ch Zerva and Lisimachos P Kondi. “Video Super-Resolution Using Plug-and-Play Priors”. In: *IEEE Access* (2024).

[16] Ségolène Martin et al. “Pnp-flow: Plug-and-play image restoration with flow matching”. In: *arXiv preprint arXiv:2410.02423* (2024).

[17] Armin Kappeler et al. “Video super-resolution with convolutional neural networks”. In: *IEEE transactions on computational imaging* 2.2 (2016), pp. 109–122.

[18] Chao Dong et al. “Learning a deep convolutional network for image super-resolution”. In: *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part IV 13*. Springer. 2014, pp. 184–199.

[19] Wenzhe Shi et al. “Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 1874–1883.

[20] Jose Caballero et al. “Real-time video super-resolution with spatio-temporal networks and motion compensation”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 4778–4787.

[21] Younghyun Jo et al. “Deep video super-resolution network using dynamic upsampling filters without explicit motion compensation”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 3224–3232.

[22] Tianfan Xue et al. “Video enhancement with task-oriented flow”. In: *International Journal of Computer Vision* 127 (2019), pp. 1106–1125.

[23] Longguang Wang et al. “Deep video super-resolution using HR optical flow estimation”. In: *IEEE Transactions on Image Processing* 29 (2020), pp. 4323–4336.

[24] Muhammad Haris, Gregory Shakhnarovich, and Norimichi Ukita. “Recurrent back-projection network for video super-resolution”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019, pp. 3897–3906.

[25] Xintao Wang et al. “Edvr: Video restoration with enhanced deformable convolutional networks”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*. 2019, pp. 0–0.

[26] Kelvin CK Chan et al. “Basicvsr: The search for essential components in video super-resolution and beyond”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2021, pp. 4947–4956.

[27] Kelvin CK Chan et al. “Basicvsr++: Improving video super-resolution with enhanced propagation and alignment”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2022, pp. 5972–5981.

[28] Ashish Vaswani et al. “Attention is all you need”. In: *Advances in neural information processing systems* 30 (2017).

[29] Jingyun Liang et al. “Swinir: Image restoration using swin transformer”. In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2021, pp. 1833–1844.

[30] Chengxu Liu et al. “Learning trajectory-aware transformer for video super-resolution”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2022, pp. 5687–5696.

[31] Jingyun Liang et al. “Recurrent video restoration transformer with guided deformable attention”. In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 378–393.

[32] Yuehan Zhang and Angela Yao. “Realviformer: Investigating attention for real-world video super-resolution”. In: *European Conference on Computer Vision*. Springer. 2024, pp. 412–428.

[33] Xi Yang et al. “Motion-guided latent diffusion for temporally consistent real-world video super-resolution”. In: *European conference on computer vision*. Springer. 2024, pp. 224–242.

[34] Shangchen Zhou et al. “Upscale-a-video: Temporal-consistent diffusion model for real-world video super-resolution”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2024, pp. 2535–2545.

[35] Xiaohui Li et al. “DiffVSR: Revealing an Effective Recipe for Taming Robust Video Super-Resolution Against Complex Degradations”. In: *arXiv preprint arXiv:2501.10110* (2025).

[36] Karol Gregor and Yann LeCun. “Learning fast approximations of sparse coding”. In: *Proceedings of the 27th international conference on international conference on machine learning*. 2010, pp. 399–406.

[37] John R Hershey, Jonathan Le Roux, and Felix Weninger. “Deep unfolding: Model-based inspiration of novel deep architectures”. In: *arXiv preprint arXiv:1409.2574* (2014).

[38] Vishal Monga, Yuelong Li, and Yonina C Eldar. “Algorithm unrolling: Interpretable, efficient deep learning for signal and image processing”. In: *IEEE Signal Processing Magazine* 38.2 (2021), pp. 18–44.

[39] Yan Yang et al. “Deep ADMM-Net for compressive sensing MRI”. In: *Proceedings of the 30th International Conference on Neural Information Processing Systems*. NIPS’16. Barcelona, Spain: Curran Associates Inc., 2016, pp. 10–18. ISBN: 9781510838819.

[40] Chris Metzler, Ali Mousavi, and Richard Baraniuk. “Learned D-AMP: Principled neural network based compressive image recovery”. In: *Advances in neural information processing systems* 30 (2017).

[41] Mark Borgerding, Philip Schniter, and Sundeep Rangan. “AMP-inspired deep networks for sparse linear inverse problems”. In: *IEEE Transactions on Signal Processing* 65.16 (2017), pp. 4293–4308.

[42] Kai Zhang et al. “Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising”. In: *IEEE transactions on image processing* 26.7 (2017), pp. 3142–3155.

[43] Kai Zhang et al. “Plug-and-play image restoration with deep denoiser prior”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44.10 (2021), pp. 6360–6376.

[44] Rita Fermanian, Mikael Le Pendu, and Christine Guillemot. “PnP-ReG: Learned regularizing gradient for plug-and-play gradient descent”. In: *SIAM Journal on Imaging Sciences* 16.2 (2023), pp. 585–613.

[45] Ping Wang et al. “Proximal Algorithm Unrolling: Flexible and Efficient Reconstruction Networks for Single-Pixel Imaging”. In: *Proceedings of the Computer Vision and Pattern Recognition Conference*. 2025, pp. 411–421.

[46] Yuqing Liu et al. “ISTA-Inspired Network for Image Super-Resolution”. In: *arXiv preprint arXiv:2210.07818* (2022).

[47] Benjamin Naoto Chiche et al. “Deep unrolled network for video super-resolution”. In: *2020 Tenth International Conference on Image Processing Theory, Tools and Applications (IPTA)*. IEEE. 2020, pp. 1–6.

[48] Seungjun Nah et al. “Ntire 2019 challenge on video deblurring and super-resolution: Dataset and study”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*. 2019, pp. 0–0.

[49] Diederik P Kingma. “Adam: A method for stochastic optimization”. In: *arXiv preprint arXiv:1412.6980* (2014).

[50] Kelvin C.K. Chan et al. “Investigating Tradeoffs in Real-World Video Super-Resolution”. In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2022.

[51] Bruce D Lucas and Takeo Kanade. “An iterative image registration technique with an application to stereo vision”. In: *IJCAI’81: 7th international joint conference on Artificial intelligence*. Vol. 2. 1981, pp. 674–679.

[52] Charles A Bouman and Gregery T Buzzard. “Generative plug and play: Posterior sampling for inverse problems”. In: *2023 59th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*. IEEE. 2023, pp. 1–7.